

### AMENDMENTS TO THE CLAIMS

1. (Currently amended): A method for preventing machine crashes due to hard errors in one of a plurality of processors in a logically partitioned data processing system, said method comprising the steps of:

detecting an error in one of said plurality of processors;

starting a deconfiguration process to deconfigure said one of said plurality of processors;

determining whether said deconfiguration process to deconfigure said one of said plurality of processors has been completed ~~deconfigured~~; [[and]]

waiting until said deconfiguration process has been completed before requesting said reboot of said system;

requesting said reboot of said system only in response to a determination that said deconfiguration process has been completed; and

rebooting only configured ones of said plurality of processors only after said deconfiguration process has been completed ~~in response to a determination that said one of said plurality of processors has been deconfigured.~~

2. (Original): The method according to claim 1, further comprising the step of in response to said detecting an error, requesting deconfiguration of said one of said plurality of processors.

3. (Original): The method according to claim 1, further comprising the step of in response to said one of said plurality of processors being deconfigured, providing an indication that said one of said plurality of processors has been deconfigured.

4. (Original): The method according to claim 1, further comprising the steps of:  
providing an exception handler routine;  
detecting, utilizing said exception handler routine, said error in said one of said plurality of processors;

determining, utilizing said exception handler routine, whether said one of said plurality of processors has been deconfigured; and  
requesting, utilizing said exception handler routine, a reboot of said configured ones of said plurality of processors.

5. (Original): The method according to claim 4, further comprising the step of requesting, utilizing said exception handler routine, said one of said processors be deconfigured.
6. (Original): The method according to claim 1, further comprising the steps of: receiving a request to deconfigure said one of said plurality of processors; deconfiguring said one of said plurality of processors; and storing an indication of deconfiguration in response to a completion of deconfiguring said one of said plurality of processors.
7. (Original): The method according to claim 6, further comprising the step of storing said indication in non-volatile memory.
8. (Original): The method according to claim 1, further comprising the steps of: providing a service processor within said logically partitioned data processing system;  
in response to said detecting an error, deconfiguring said one of said plurality of processors utilizing said service processor; and  
providing, utilizing said service processor, an indication that said one of said plurality of processors has been deconfigured.
9. (Original): The method according to claim 1, wherein the step of detecting an error in one of said plurality of processors further comprises the step of detecting a hard error in one of said plurality of processors.

10. (Original): The method according to claim 1, wherein the step of detecting an error in one of said plurality of processors further comprises the step of detecting an address translation error in one of said plurality of processors.
11. (Original): The method according to claim 1, wherein the step of detecting an error in one of said plurality of processors further comprises the step of detecting a translation look aside buffer address translation error in one of said plurality of processors.
12. (Original): The method according to claim 1, wherein the step of detecting an error in one of said plurality of processors further comprises the step of detecting a data effective to real address translation error in one of said plurality of processors.
13. (Original): A method for preventing machine crashes due to hard errors in one of a plurality of processors in a logically partitioned data processing system, said method comprising the steps of:
- providing an error handler routine;
  - providing a service processor within said logically partitioned data processing system;
  - detecting, utilizing said error handler routine, an error in one of said plurality of processors;
  - in response to said error handler routine detecting an error, requesting, utilizing said error handler routine, said service processor to deconfigure said one of said plurality of processors;
  - deconfiguring, utilizing said service processor, said one of said plurality of processors;
  - in response to a completion of said deconfiguration of said one of said plurality of processors, storing, utilizing said service processor, an indication in a non-volatile memory that said one of said plurality of processors has been deconfigured;

determining, utilizing said error handler routine, whether an indication has been stored in said non-volatile memory indicating that said one of said plurality of processors has been deconfigured;

only in response to a determination that said indication has been stored in said non-volatile memory that said one of said plurality of processors has been deconfigured, requesting, utilizing said error handler routine, a reboot of configured ones of said plurality of processors; and

rebooting, utilizing said service processor, said configured ones of said plurality of processors.

14. (Currently amended): A logically partitioned data processing system for preventing machine crashes due to hard errors in one of a plurality of processors included in said logically partitioned data processing system, comprising:

means for detecting an error in one of said plurality of processors;

means for starting a deconfiguration process to deconfigure said one of said plurality of processors;

means for determining whether said deconfiguration process to deconfigure said one of said plurality of processors has been completed deconfigured; [[and]]

means for waiting until said deconfiguration process has been completed before requesting said reboot of said system;

means for requesting said reboot of said system only in response to a determination that said deconfiguration process has been completed; and

means for rebooting only configured ones of said plurality of processors only after said deconfiguration process has been completed in response to a determination that said one of said plurality of processors has been deconfigured.

15. (Original): The system according to claim 14, further comprising in response to said detecting an error, means for requesting deconfiguration of said one of said plurality of processors.

16. (Original): The system according to claim 14, further comprising in response to said one of said plurality of processors being deconfigured, means for providing an indication that said one of said plurality of processors has been deconfigured.
17. (Original): The system according to claim 14, further comprising:  
an exception handler routine;  
said exception handler routine for detecting said error in said one of said plurality of processors;  
said exception handler routine for determining whether said one of said plurality of processors has been deconfigured; and  
said exception handler routine for requesting a reboot of said configured ones of said plurality of processors.
18. (Original): The system according to claim 17, further comprising said exception handler routine for requesting said one of said processors be deconfigured.
19. (Original): The system according to claim 14, further comprising:  
means for receiving a request to deconfigure said one of said plurality of processors;  
means for deconfiguring said one of said plurality of processors; and  
means for storing an indication of deconfiguration in response to a completion of deconfiguring said one of said plurality of processors.
20. (Original): The system according to claim 19, further comprising a non-volatile memory for storing said indication.
21. (Original): The system according to claim 14, further comprising:  
a service processor within said logically partitioned data processing system;  
in response to said detecting an error, said service processor for deconfiguring said one of said plurality of processors; and

said service processor for providing an indication that said one of said plurality of processors has been deconfigured.

22. (Original): The system according to claim 14, wherein said means for detecting an error in one of said plurality of processors further comprises means for detecting a hard error in one of said plurality of processors.

23. (Original): The system according to claim 14, wherein said means for detecting an error in one of said plurality of processors further comprises means for detecting an address translation error in one of said plurality of processors.

24. (Original): The system according to claim 14, wherein said means for detecting an error in one of said plurality of processors further comprises means for detecting a translation look aside buffer address translation error in one of said plurality of processors.

25. (Original): The system according to claim 14, wherein said means for detecting an error in one of said plurality of processors further comprises means for detecting a data effective to real address translation error in one of said plurality of processors.

26. (Currently amended): A system for preventing machine crashes due to hard errors in one of a plurality of processors in a logically partitioned data processing system, said system comprising:

an error handler routine;

a service processor within said logically partitioned data processing system;

said error handler routine for detecting an error in one of said plurality of processors;

in response to said error handler routine detecting an error, said error exception handler routine for requesting said service processor to deconfigure said one of said plurality of processors;

said service processor for deconfiguring said one of said plurality of processors;

in response to a completion of said deconfiguration of said one of said plurality of processors, said service processor for storing an indication in a non-volatile memory that said one of said plurality of processors has been deconfigured;

said error handler routine for determining whether an indication has been stored in said non-volatile memory indicating that said one of said plurality of processors has been deconfigured;

only in response to a determination that said indication has been stored in said non-volatile memory that said one of said plurality of processors has been deconfigured, said exception handler routine for requesting a reboot of configured ones of said plurality of processors; and

said service processor for rebooting said configured ones of said plurality of processors.

27. (Currently amended): A computer program product for preventing machine crashes due to hard errors in one of a plurality of processors in a logically partitioned data processing system, said computer program product comprising:

instruction means for detecting an error in one of said plurality of processors;

instruction means for starting a deconfiguration process to deconfigure said one of said plurality of processors;

instruction means for determining whether said deconfiguration process to deconfigure said one of said plurality of processors has been completed ~~deconfigured~~;

[[and]]

instruction means for waiting until said deconfiguration process has been completed before requesting said reboot of said system;

instruction means for requesting said reboot of said system only in response to a determination that said deconfiguration process has been completed; and

instruction means for rebooting only configured ones of said plurality of processors only after said deconfiguration process has been completed ~~in response to a determination that said one of said plurality of processors has been deconfigured.~~

28. (Original): The product according to claim 27, further comprising in response to said detecting an error, instruction means for requesting deconfiguration of said one of said plurality of processors.
29. (Original): The product according to claim 27, further comprising in response to said one of said plurality of processors being deconfigured, instruction means for providing an indication that said one of said plurality of processors has been deconfigured.
30. (Original): The product according to claim 27, further comprising:  
instruction means for providing an exception handler routine;  
instruction means for detecting, utilizing said exception handler routine, said error in said one of said plurality of processors;  
instruction means for determining, utilizing said exception handler routine, whether said one of said plurality of processors has been deconfigured; and  
instruction means for requesting, utilizing said exception handler routine, a reboot of said configured ones of said plurality of processors.
31. (Original): The product according to claim 30, further comprising instruction means for requesting, utilizing said exception handler routine, said one of said processors be deconfigured.
32. (Original): The product according to claim 27, further comprising:  
instruction means for receiving a request to deconfigure said one of said plurality of processors;  
instruction means for deconfiguring said one of said plurality of processors; and  
instruction means for storing an indication of deconfiguration in response to a completion of deconfiguring said one of said plurality of processors.
33. (Original): The product according to claim 32, further comprising instruction means for storing said indication in non-volatile memory.



34. (Original): The product according to claim 27, further comprising:  
instruction means for providing a service processor within said logically partitioned data processing system;  
in response to said detecting an error, instruction means for deconfiguring said one of said plurality of processors utilizing said service processor; and  
instruction means for providing, utilizing said service processor, an indication that said one of said plurality of processors has been deconfigured.
35. (Original): The product according to claim 27, wherein said instruction means for detecting an error in one of said plurality of processors further comprises instruction means for detecting a hard error in one of said plurality of processors.
36. (Original): The product according to claim 27, wherein said instruction means for detecting an error in one of said plurality of processors further comprises instruction means for detecting an address translation error in one of said plurality of processors.
37. (Original): The product according to claim 27, wherein said instruction means for detecting an error in one of said plurality of processors further comprises instruction means for detecting a translation look aside buffer address translation error in one of said plurality of processors.
38. (Original): The product according to claim 27, wherein said instruction means for detecting an error in one of said plurality of processors further comprises instruction means for detecting a data effective to real address translation error in one of said plurality of processors.
39. (Original): A computer program product for preventing machine crashes due to hard errors in one of a plurality of processors in a logically partitioned data processing system, said computer program product comprising:  
instruction means for providing an error handler routine;

instruction means for providing a service processor within said logically partitioned data processing system;

instruction means for detecting, utilizing said error handler routine, an error in one of said plurality of processors;

in response to said error handler routine detecting an error, instruction means for requesting, utilizing said error handler routine, said service processor to deconfigure said one of said plurality of processors;

instruction means for deconfiguring, utilizing said service processor, said one of said plurality of processors;

in response to a completion of said deconfiguration of said one of said plurality of processors, instruction means for storing, utilizing said service processor, an indication in a non-volatile memory that said one of said plurality of processors has been deconfigured;

instruction means for determining, utilizing said error handler routine, whether an indication has been stored in said non-volatile memory indicating that said one of said plurality of processors has been deconfigured;

only in response to a determination that said indication has been stored in said non-volatile memory that said one of said plurality of processors has been deconfigured, instruction means for requesting, utilizing said error handler routine, a reboot of configured ones of said plurality of processors; and

instruction means for rebooting, utilizing said service processor, said configured ones of said plurality of processors.